Lab 1: Fun with Collections

Due: 2016/05/12 23:59:59

Weilun Xiong (azardf4yy@gmail.com)

History

- From Stanford CS106B/X
  (http://web.stanford.edu/class/cs106b/assignments.html)

- Linux
- g++
- Stanford C++ Library

History

- From Stanford CS106B/X
  (http://web.stanford.edu/class/cs106b/assignments.html)

- Linux
- g++
- Stanford C++ Library

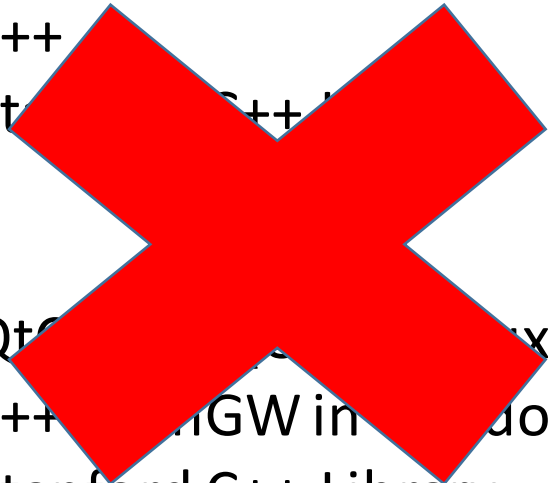- QtCreator (OS X, Linux, Windows)
- g++ (MinGW in Windows)
- Stanford C++ Library

History

- From Stanford CS106B/X
  (http://web.stanford.edu/class/cs106b/assignments.html)

- Linux
- g++
- St~~andard C++ L~~

- Qt~~Creator (Lin~~ux, Windows)
- g++~~ (Mi~~nGW in ~~Win~~dows)
- Stanford C++ Library

- Windows (recommend)
  OS X, Linux

- Visual C++ (recommend)
  g++, clang

- STL

Simple, naïve, but useful

Target

- Practice C++ programming skill
- Learn the power of data structure
  - Solving interesting reality problem
- Part I : Word Ladders
- Part II: Random Writer

- STL Documents:
  - http://www.cplusplus.com/
  - http://zh.cppreference.com/w/cpp

Part I : Word Ladders

- A kind of word game (https://en.wikipedia.org/wiki/Word_ladder)

  COLD → CO**R**D → C**A**RD → **W**ARD → WAR**M**
  COLD → CO**R**D → COR**M** → **W**ORM → W**A**RM
  COLD → **W**OLD → WO**R**D → W**A**RD → WAR**M**

- Your task:
  Develop a tool, input start and target word,
  automate show the path

- Demo

- If you have this tool when playing with friends……

Part I : Word Ladders

- 100+- lines C++ code
- BFS (Breadth first search)

Recommend data structure:
```
set<string> english;
set<string> used_words;
queue<vector<string>> search_queue;
```

{cat}

{~~{cat}~~, {c<u>o</u>t}, {ca<u>d</u>}, {ca<u>r</u>}}

{~~{cot}~~, {cad}, {car}, **{cot,<u>d</u>ot}**, **{cot,cog}**, **{cot,co<u>n</u>}**}

{~~{cad}~~, {car}, {cot,dot}, {cot,cog}, {cot,con}, **{cad,<u>b</u>ad}**}

{~~{car}~~, {cot,dot}, {cot,cog}, {cot,con}, {cad,bad}, **{car,<u>b</u>ar}**, **{car,<u>w</u>ar}**}

{~~{cot,dot}~~, {cot,cog}, {cot,con}, {cad,bad}, {car,bar}, {car,war}, **{cot,dot,dog}**}

Part I : Word Ladders

Question

Part II : Random Writer

- SCIgen
  - https://pdos.csail.mit.edu/archive/scigen/
  - http://www.nowcoder.com/discuss/206

- Input: "The four style of writing about beans flavored with aniseed"
- Output: "Writing style analysis based on beans flavored with aniseed"
- Demo

Part II : Random Writer

- 100+- lines C++ code
- Markov Model

Recommend data structure:
```
map<string, vector<char> > model;
```

```
Order = 5
Input: "p-p-p-poker face, p-p-poker face"
Model: = {
    "p-p-p": ['-', 'o', '-', 'o'],
    "-p-p-": ['p', 'p'],
    "-p-po": ['k', 'k'],
    "p-pok": ['e', 'e'],
    "-poke": ['r', 'r'],
    "poker": [' ', ' '],
    ……
}
```

Part II : Random Writer

- 100+- lines C++ code
- Markov Model

Recommend data structure:
```
map<string, vector<char> > model;
```

```
Order = 5
Model = {
    "hello": [' ', ' ', ' ', 'a'],
    "ello ": ['w', 'w', 'w', 'w', 'w'],
    "llo w": ['o', 'o', 'o', 'a']
    ……
}

Generate: "hello" -> "hello " -> "hello w" -> "hello wo"
```

Part II : Random Writer

Question

Just Read

Lab 1: Fun with Collections

开始时间：2016/04/21 00:00:00
截止时间：2016/05/12 23:59:59

负责助教：熊伟伦 (azardf4yy@gmail.com)
答疑时间：Lab1可提交时间内的每周三、五下午2点到4点，软院5404，熊伟伦

- 任务详细描述：lab1.pdf
- 初始文件和Demo：lab1_handout.zip
- Slides：4月24号习题课后放出